REMARKS/ARGUMENTS

In the Office Action, the Examiner noted that claims 1-25 are pending in the application. The Examiner additionally stated that claims 1-25 are rejected. By this communication, claims 1, 8, 15-16, and 21 are amended. Hence, claims 1-25 are pending in the application.

Applicant hereby requests further examination and reconsideration of the application, in view of the foregoing amendments.

In the Specification

Applicant has amended the specification to secure a substantial correspondence between the claims amended herein and the remainder of the specification. No new matter is presented.

In the Claims

Objections

The Examiner objected to claim 15 because it is believed that portions deleted from the claim were done so unintentionally. In reply, Applicant notes that the amendment was correct in that the cryptography unit, which was formerly recited in claim 15, was added into the language of claim 1, and what remained as a limitation in claim 15 was "wherein said cryptography unit executes said plurality of cryptographic rounds on each of a plurality of input text blocks to generate a corresponding each of a plurality of output text blocks, wherein said plurality of cryptographic rounds are prescribed by a control word that is provided to said cryptography unit."

Accordingly, it is requested that the objection to claim 15 be withdrawn.

Rejections Under 35 U.S.C. §103(a)

The Examiner rejected claims 1-25 under 35 U.S.C. 103(a) as being unpatentable over Kessler, US6789147 (hereinafter, "Kessler"), in view of Miller, US6081884 (hereinafter, "Miller"). Applicant respectfully traverses the Examiner's rejections.

Referring to claims 1 and 21, the Examiner noted that Kessler discloses a co-processor that includes multiple execution units (Figure 2) wherein each of the execution units

includes an execution queue to store cryptographic instructions received by the co-processor (Figure 8). The Examiner noted that this meets the limitation of a fetch logic, disposed within a microprocessor, configured to receive a cryptographic instruction as a part of an instruction flow executing on said microprocessor, wherein said cryptographic instruction prescribes one of the cryptographic operations.

The Examiner also stated that the execution units include a plurality of operation blocks that correspond to different cryptographic operations that are used depending upon the type of instruction received in the execution queue (Figure 8 & Col. 9, lines 7-43), which meets the limitation of wherein said cryptographic instruction prescribes one of a plurality of cryptographic algorithms, algorithm logic, operatively coupled to said cryptographic instruction, configured to direct said microprocessor to execute said one of the cryptographic operations according to said one of a plurality of cryptographic algorithms.

The Examiner further observed that using the appropriate operation block, the corresponding cryptographic algorithm is used when processing the received instruction (Col. 9, lines 28-43), which meets the limitation of execution logic, operatively coupled to said algorithm logic, configured to execute said one of the cryptographic operations.

The Examiner also noted that The operation blocks correspond to cryptographic algorithms such as AES, 3DES, DES, and RC4 (Figures 5 & 8), which meets the limitation of executing a plurality of cryptographic rounds required to complete said one of the cryptographic operations.

The Examiner stated that Kessler does not specify that the co-processor utilizes the x86 instruction set, however, it would have been obvious to one of ordinary skill in the art at the time the invention was made for the co-processor described in Kessler to implement the x86 instruction set because the x86 instruction set has been widely accepted because of it's compatibility with a large amount of software as taught by Miller (Col. 2, lines 9-14). The Examiner added that Applicant's specification shows that integer instructions are inherent to the x86 instruction set (Page 27), and therefore, when implementing the

x86 instruction set in the co-processor of Kessler, as previously described, the execution units would effectively operate as a "integer unit" as claimed.

In reply to Applicant's arguments and remarks made in the previous communication, the Examiner stated that Applicant's assertion that "Kessler clearly teaches that his device is a coprocessor, and not a microprocessor, thus Kessler does not meet the aforementioned limitation," is not persuasive because the definition, as defined by dictionary.com, of a coprocessor is "a microprocessor that performs specialized functions that the central processing unit cannot perform or cannot perform as well and as quickly."

In reply to Applicant's remark that "Kessler's execution units do not include an integer unit," the Examiner agreed that Kessler does not specify an integer unit per so, but that as was stated in the Office Action mailed 13 November 2007, on page 9, that it would have been obvious to one of ordinary skill in the art at the time the invention was made for the co-processor described in Kessler to implement the x86 instruction set because the x86 instruction set has been widely accepted because of it's compatibility with a large amount of software as taught by Miller (Col. 2, lines 9-14), and that Applicant's specification shows that integer instructions are inherent to the x86 instruction set (Page 27). Therefore, when implementing the x86 instruction set in the co-processor of Kessler, as previously described, the execution units would effectively operate as a "integer unit" as claimed.

Applicant refers the Examiner to arguments made in previous responses and to the substance of the Examiner interview which took place on 04/15/2008. To summarize, throughout prosecution of the instant application, it has been the Examiner's assertion that the terms processor, co-processor, microprocessor, and host processor are entirely equivalent. Hence, according to the Examiner, the "co-processor" of Kessler would be equivalent to the "microprocessor" of the instant claims because such are simply words that are employed to express the same meaning. To support the Examiner's assertion, in the instant Office Action, a "definition" from dictionary.com is quoted, which is that a "coprocessor" "is a microprocessor that performs specialized functions that the central processing unit cannot perform or cannot perform as well and as quickly." It is from this

definition that the Examiner relies upon to equate the terms "coprocessor" and "microprocessor."

Yet, an analogous definition would be, for example, that an "apple" is a "fruit" that is red, hangs from a tree, and keeps the doctor away if eaten once per day. But from this "definition" would the Examiner agree that the terms "apple" and "fruit" are equivalent? Applicant respectfully asserts that the Examiner would not concur that the terms are equivalent. In like fashion, neither are the terms "co-processor" and "microprocessor." In fact, Applicant would argue that one skilled in the art would concur that what is referred to in the present application as a "coprocessor" is entirely synonymous with the "coprocessor" of Kessler, and that the present application's use of the term "microprocessor" is equivalent to Kessler's "host processor" with the caveat that a microprocessor is presently contained within a single die on an integrated circuit.

Moreover, Applicant would respectfully submit that a microprocessor, as used conventionally in the art and in the present application, is substantially equivalent to what was formerly known as a "central processing unit" (CPU). In the Examiner's dictionary.com quote, a coprocessor is limited in that it only performs specialized functions that the CPU cannot perform or cannot perform as well or as quickly.

In the background of the instant application, this limitation is specifically pointed out in terms of today's microprocessor's (CPU's) not being able to perform cryptographic operations as well or as quickly as desired.

Applicant's use of the term microprocessor is entirely consistent with that of a CPU, and it is respectfully asserted that the term "CPU" is less commonly used today than the term "microprocessor" for a CPU may be configured of several discrete components, but a microprocessor is in fact a CPU on a chip. Applicant again asserts that such is the reason the present application does not teach a CPU, but rather a microprocessor, for the embodiments according to the present invention are all disclosed in terms of a "microprocessor," that is, a single chip.

Hence, it is readily concurred with by those in the art that a "coprocessor' is a "limited microprocessor" for it only performs supplemental functions. Kessler's coprocessor is a

prime example of a coprocessor being handed off an encryption function from his host processor.

In the Interview of 04/15/2008, the Examiner suggested that the addition of structural limitations to the claims may well provide the desired distinction between the coprocessor of Kessler and the microprocessor according to the present invention. Applicant appreciates this suggestion and notes that such structural distinctions were previously provided as limitations in claims 1, 16, and 21, but the instant Office action did not address these distinctions. More specifically, claims 1, 16, and 21 were amended to recite that the cryptographic instruction "is one of the instructions in an application program being executed by said microprocessor." That is, as Applicant pointed out in previous responses, a coprocessor only executes threads, or single instructions, or single tasks, handed off by a host microprocessor. In contrast, a microprocessor (i.e., CPU) executes application programs. This is one of the major distinctions between a coprocessor and a microprocessor fetches the instructions from system memory for the application program and the program is executed by the microprocessor. It is only instruction threads that are executed by a coprocessor. To support this assertion the execution of an application program is a capability inherent in a microprocessor (CPU) and not a coprocessor, the following definition from Wikipedia.com is submitted.

"A Central Processing Unit (CPU), or sometimes just called *processor*, is a description of a class of logic machines that can *execute computer programs*. This broad definition can easily be applied to many early computers that existed long before the term "CPU" ever came into widespread usage. The term itself and its initialism have been in use in the computer industry at least since the early 1960s (Weik 1961). The form, design and implementation of CPUs have changed dramatically since the earliest examples, but their fundamental operation has remained much the same." [Emphasis provided]

Applicant notes that one of the capabilities which the above definition imbues to a processor is that it executes computer programs. And it is respectfully asserted that a coprocessor cannot execute a computer program. Applicant has referred to a computer

program in the instant application as both an application program to and an operating system, to distinguish between the two classes of computer programs.

Consequently, to distinguish the microprocessor according to the present invention over the coprocessor of Kessler, Applicant amended claims 1, 16, and 21 in the previous response to recite "wherein said cryptographic instruction is one of the instructions in an application program being executed by said microprocessor." Applicant at this point realizes that the Examiner may have construed the above statement to infer that it is the cryptographic instruction itself that is executed by the microprocessor and that the remainder of the application program may be executed by another device. Accordingly, claims 1, 16, and 21 are amended by this communication to specifically recite "wherein said cryptographic instruction is one of the instructions in an application program, wherein said application program is executed by said microprocessor to obtain expected results."

The above amendments recite that the microprocessor according to the present invention executes the application program, one instruction of which is the cryptographic instruction, and therefore without ambiguity, the microprocessor according to the present invention possesses the capability to execute application programs, which is a capability not provided for by a coprocessor.

In reference to claims 1 and 21, the Examiner's assertion that Kesslers's disclosure of a co-processor having multiple execution units meets the limitation of a fetch logic, . . . , configured to receive a cryptographic instruction as part of an instruction flow," may very well apply to an instruction thread such as is conventionally offloaded to a co-processor. However, the substantive recitation of the claims, in both the previous amendment and in the instant amendment, relates to the execution of an *application program* and not to an instruction flow. Applicant has searched the previous office actions and Kessler as well to find where it may be inferred that Kessler's special purpose coprocessor may be capable of executing application programs. As a result, Applicant finds that the Examiner did not address this limitation, but rather spoke to execution of an instruction

flow. And Kessler certainly teaches only the execution of a specialized function that has been handed off from a host processor.

Regarding the teachings of Miller, Applicant indeed agrees that the x86 instruction set has been widely accepted because of it's compatibility with a large amount of software. However, Miller in no way suggests that a microprocessor be provided that is both x86 compatible and that has an x86 cryptographic instruction that takes advantage of an integral cryptographic unit. Moreover, Kessler does not suggest such a device either. By combining the two references, one skilled in the art would be led to conclude that Kessler's coprocessor may be useful in an x86 environment because it could offload cryptographic functions which would otherwise have to be performed via operating system intensive subroutine calls.

Based upon the above arguments and instant amendments, Applicant respectfully requests that the rejection of claim 1 be withdrawn.

Claim 21 recites substantially the same limitations as have been argued above as being allowable over Kessler, Miller, or a combination of the two references. Accordingly, it is requested that the rejection of claim 21 be withdrawn as well

Furthermore, the Examiner's observation that it would have been obvious to one of ordinary skill in the art at the time the invention was made for the co-processor described in Kessler to implement the x86 instruction set because the x86 instruction set has been widely accepted because of it's compatibility with a large amount of software as taught by Miller (Col. 2, lines 9-14), Applicant finds—respectfully—ludicrous and unfounded. Such an observation that the functionality of literally hundreds of extant instructions be added to a specialized encryption coprocessor that only possesses a few primitive functions such as Kessler's is not a reasonable conclusion to make whatsoever. Clearly, Kessler teaches that cryptographic functions are better offloaded to a coprocessor.

With respect to claims 2-15, these claims depend from claim 1 and add further limitations that are neither anticipated nor made obvious by Kessler, Miller, or a combination of the two references. Accordingly, Applicant respectfully requests that the Examiner withdraw the rejections of claims 2-15.

With respect to claims 22-25, these claims depend from claim 21 and add further limitations that are neither anticipated nor made obvious by Kessler, Miller, or a combination of the two references. Accordingly, Applicant respectfully requests that the Examiner withdraw the rejections of claims 22-25.

As per claim 16, the Examiner noted that Kessler discloses a co-processor that includes multiple execution units (Figure 2) wherein each of the execution units includes an execution queue to store cryptographic instructions received by the co-processor (Figure 8), which meets the limitation of a cryptographic unit within a microprocessor, configured to execute one of the cryptographic operations response to receipt of a cryptographic instruction that prescribes said one of the cryptographic operations, wherein said cryptographic instruction is one of the instructions in an application program that are fetched from memory by fetch logic in said microprocessor. The Examiner additionally observed that the execution units include a plurality of operation blocks that correspond to different cryptographic operations that are used depending upon the type of instruction received in the execution queue (Figure 8 & Col. 9, lines 7-43), which meets the limitation of an algorithm field, configured to prescribed one of a plurality of cryptographic algorithms to be employed when executing said one of the cryptographic operations. The Examiner noted that using the appropriate operation block, the corresponding cryptographic algorithm is used when processing the received instruction (Col. 9, lines 28-43), which meets the limitation of algorithm logic, operatively coupled to said cryptography unit, configured to direct said device to perform said one of the cryptographic operations according to said one of the plurality of cryptographic algorithms.

Applicant respectfully disagrees and directs the Examiner's attention to arguments provided above in traversal of the rejections of claims 1 and 21. More specifically, claim 16, as amended herein, recites, *inter alia,* wherein said microprocessor executes said application program to obtain expected results. As has been argued above, Kessler's coprocessor is incapable of executing an application program. It only executes cryptographic functions offloaded from the host processor.

Since these limitations are not taught, contemplated, or suggested by Kessler, Miller, or a combination of the two references, it is requested that the rejection of claim 16 be withdrawn.

With respect to claims 17-20, these claims depend from claim 16 and add further limitations that are neither anticipated nor made obvious by Kessler, Miller, or a combination of the two references. Accordingly, Applicant respectfully requests that the Examiner withdraw the rejections of claims 16-20.

CONCLUSIONS

Applicant believes this to be a complete response to all of the issues raised in the instant office action and further submits, in view of the amendments and arguments advanced above, that claims 1-25 are in condition for allowance. Reconsideration of the rejections is requested, and allowance of the claims is solicited.

Applicant also notes that any amendments made by way of this response, and the observations contained herein, are made solely for the purpose of expediting the patent application process in a manner consistent with the PTO's Patent business Goals (PBG), 65 Fed. Reg. 54603 (September 8, 2000), and are furthermore made without prejudice to Applicant under this or any other jurisdictions. It is moreover asserted that insofar as any subject matter might otherwise be regarded as having been abandoned or effectively disclaimed by virtue of amendments made herein and/or incorporated in attachments submitted with this response, Applicants wishes to reserve the right and hereby provides notice of intent to restore such subject matter and/or file a continuation application in respect thereof.

Applicant earnestly requests that the Examiner contact the undersigned practitioner by telephone if the Examiner has any questions or suggestions concerning this amendment, the application, or allowance of any claims thereof.

Respectfully submitted,
HUFFMAN PATENT GROUP, LLC

/Richard K. Huffman/

By: _____

RICHARD K. HUFFMAN, P.E.
Registration No. 41,082
Tel: (719) 575-9998

01/08/2008

Date: _____